electronics

MDPI

Article

D-UAP: Initially Diversified Universal Adversarial Patch Generation Method

Lei Sun [1], Xiaoqin Wang [1,*], Youhuan Yang [2] and Xiuqing Mao [1]

[1] Three Academy, PLA Information Engineering University, Zhengzhou 450000, China
[2] Software Institute, Zhengzhou University, Zhengzhou 450000, China; 202012332015247@gs.zzu.edu.cn
* Correspondence: xxgcdx1502@163.com

Abstract: With the rapid development of adversarial example technologies, the concept of adversarial patches has been proposed, which can successfully transfer adversarial attacks to the real world and fool intelligent object detection systems. However, the real-world environment is complex and changeable, and the adversarial patch attack technology is susceptible to real-world factors, resulting in a decrease in the success rate of attack. Existing adversarial-patch-generation algorithms have a single direction of patch initialization and do not fully consider the impact of initial diversification on its upper limit of adversarial patch attack. Therefore, this paper proposes an initial diversified adversarial patch generation technology to improve the effect of adversarial patch attacks on the underlying algorithms in the real world. The method uses YOLOv4 as the attack model, and the experimental results show that the attack effect of the adversarial-patch-attack method proposed in this paper is higher than the baseline 8.46%, and it also has a stronger attack effect and fewer training rounds.

Keywords: object detection; YOLOv4; adversarial attack; adversarial patch; output diversification initialization

Citation: Sun, L.; Wang, X.; Yang, Y.; Mao, X. D-UAP: Initially Diversified Universal Adversarial Patch Generation Method. *Electronics* **2023**, *12*, 3080. https://doi.org/10.3390/electronics12143080

Academic Editor: Silvia Liberata Ullo

Received: 26 May 2023
Revised: 30 June 2023
Accepted: 30 June 2023
Published: 14 July 2023

1. Introduction

In recent years, with the rapid development of deep learning technology, deep learning models represented by convolutional neural networks have been widely used in various computer vision applications, such as image classification [1,2], object detection [3–5], face recognition [6–8], object tracking [9–11], etc. As one of the basic tasks of computer vision, the object detection task has achieved breakthroughs with the continuous development of deep learning, and the detection method has changed from traditional manual setting feature recognition [12,13] to automatic feature extraction based on neural networks, which greatly improves the performance of object detection. At present, the mainstream object detection model can be divided into two types according to its detection stage: one-stage and two-stage detection models. The YOLO series model, represented by YOLOv4, is a one-stage model; it realizes end-to-end work, and the positioning task is carried out simultaneously with the regression task, which leads to a faster detection speed and is suitable for real-time object detection tasks. The two-stage detection model [14,15] divides the detection task into two stages; the model first identifies the possible target locations to generate area suggestions and then classifies and identifies these area suggestions. Compared with traditional detection systems, the detection model based on a deep neural network shows good detection performance and detection speed.

The object detection model based on deep learning obtains excellent performance brought by deep neural networks, but it also inherits the shortcomings of neural networks, i.e., it is vulnerable to adversarial example attacks. Adversarial examples are samples that can be made to obtain unexpected results (e.g., classification errors and failed pedestrian detectors) by adding special perturbations that cannot be recognized by the human eye. With the deepening of research in the field of image classification, researchers have shifted

their eyes to more complex object detection tasks. The object detection system adopts a preset prior frame to draw a bounding box at the target position to locate objects and identify categories. The number of targets that need to be attacked is much larger than that of image classification, so its attack is more complex than the image classification task. Xie [16] extended the adversarial examples of image classification to object detection and proposed a DAG method, which assigns adversarial labels to regions and optimizes the overall loss function. Lu [7] proposed a stop-sign vanishing attack for videos and successfully attacked the faster region-based convolutional neural network (Faster RCNN) method [15] using the fast gradient sign method (FGSM) [17], which proved the effectiveness of object detection adversarial attack. Although the adversarial example based on global perturbation can successfully attack the object detection and recognition system, this attack method of adding perturbation to the global image cannot be transferred to the physical world.

To obtain an attack adversarial example that can be successfully transferred to the physical world, the concept of adversarial patch [18] was proposed. Li [19] attacked the object detection system for the first time by adding patches to the image and achieved certain results, and the adversarial patch well met the need of transferring adversarial examples to physical world attacks. At present, combined with the actual mainstream tasks, the research on adversarial patch attacks mainly focuses on three task areas: evading face recognition systems, pedestrian detection systems, and automatic driving systems that detect stop signs. Different from the single detection category of the facial recognition system and the fixed stop sign mode in the autonomous driving system, Thys [20] believed that it is more challenging to attack within a single category of 'people' in the object detection task and proposed an adversarial-patch-generation method to successfully evade the detection of the YOLOv2 object detector. Based on this, the literature [21] presented a printable adversarial T-shirt, and the above research further verified the effectiveness of adversarial masks in the physical world. Hu et al. [22] further combined cutting techniques to generate multi-angle detection-resistant T-shirts.

Although adversarial patches have been extensively studied in wearable and multi-angle attacks, the existing adversarial patch generation methods adopt a single initialization method and do not consider the initial diversification of adversarial patches, resulting in the upper limit of adversarial patch attacks in the subsequent training process. Meanwhile, most of the existing adversarial patch attack research is carried out on the YOLOv2 object detector, but with the continuous development of detection technology, the YOLOv4 detector achieves better detection performance and is more widely used, which has a more advanced network architecture and stronger detection performance than YOLOv2. The original creator of YOLOv3 no longer updates the system after updating YOLOv3, and since then, only Alexey's improved YOLOv4 has been recognized. At present, the YOLOv4 model is widely used in some specific places because of high accuracy. Therefore, to generate an adversarial T-shirt with a stronger attack effect in the actual process, this paper selects YOLOv4 as the attack model and proposes a diversified initial method based on the existing classical adversarial-patch-generation algorithm to further improve the avoidance effect of the adversarial patch on pedestrian detectors.

In summary, the main contributions of this paper are summarized as follows:

1. Based on the upper limit of attacks caused by single-adversarial-patch initialization, this paper proposes an initial diversified-attack method, which is 8.46% higher than the classical adversarial-patch-attack-effect method on the *INRIA* dataset.
2. Based on the idea of diversifying the initial direction of adversarial patches, the adversarial example generation is faster than the original single-direction training, saving an average of 300 training rounds.

2. Related Work

This section introduces the related work, which is divided into two aspects: object detection model and adversarial patch technology. Firstly, the research background of the YOLO series is provided, and the basic framework and the improved structure of the test

model YOLOv4 model used in this paper are presented. Secondly, the relevant research fields and research status of adversarial patch are presented, and the limitations of the single initial diversification of existing adversarial patch methods are described.

2.1. Object Detector

YOLO is a single-stage detection model that redefines the detection task as a single regression problem, thus enabling end-to-end detection directly from image pixels to bounding box coordinates and class probabilities. Therefore, the YOLO detection model is not used for two-stage detection models with a faster detection speed. YOLOv1 [23], proposed by Joseph in 2016, divides the image into SS grids to detect the object in the center of the real box. YOLOv2 [24] was optimized on YOLOv1 in 2017, the backbone network was slightly adjusted, the fully convolutional network architecture was adopted, and multi-scale training was introduced to improve the generalization ability and detection effect of the network. However, YOLOv2 does not achieve good detection performance for small targets, and the swarm detection effect is not satisfactory. Then, YOLOv3 [4] was further optimized. It uses Darknet53 as the network backbone, adopts cross-scale feature fusion, and selects the anchor size obtained by clustering on the MS COCO dataset. These optimizations improve the detection performance of YOLOv3 for small targets, but its recall is low, and the population detection performance is poor. Later, Alexey proposed an improved version of YOLOv4 [3], and the main improvement is that the network structure adopts CSPDarknet53 as the backbone network, which solves the problem of large computation in inference from the perspective of structural design, enhances the learning ability of CNN, maintains light weight and reduces memory costs, and uses SPP layers to ensure uniform dimensions for output. The PAN path aggregation network is adopted to enhance from the bottom up, making it easier for low-level spatial information to propagate to the top. Moreover, Mosaic data augmentation, Mish activation function, etc., are exploited to enhance model robustness. Subsequently, different researchers proposed other YOLO detection models based on this literature [5,25,26], and they paid more attention to the detection of lightweight and industrial-specific tasks. YOLOv7 is a representative lightweight detection model with excellent performance, and it will be widely studied.

In this paper, YOLOv4 is chosen as the test attack model because it has been widely studied and has good performance; its structure is shown in Figure 1. When the YOLO series model detects objects from an image, it first divides the image into grids of different sizes, each grid being responsible for a different area. YOLOv4 object detection model divides the image into three grids of different sizes to detect targets of different sizes, and each grid is based on the preset priori box size, thereby generating three prior boxes of different sizes as shown in Figure 2. For each prior box, the model directly outputs the corresponding probability, position adjustment parameters, and class probability of the object, such as the adjustment parameters of the prior box (x, y, w, h), the confidence score $conf$, and the probability of the target category in the prior box $(P_c ls1, P_c ls2, P_c ls3 \ldots P_c lsn)$. Each grid point contains a total of 5 + n_classes-bit parameters, including 4-bit adjustment parameters, 1 position reliability parameter, and the number of data categories n_classes. Finally, the detection bounding boxes with the highest confidence are selected as the result output by non-maximum suppression, and the final output detection bounding boxes in red color in Figure 1 are obtained.

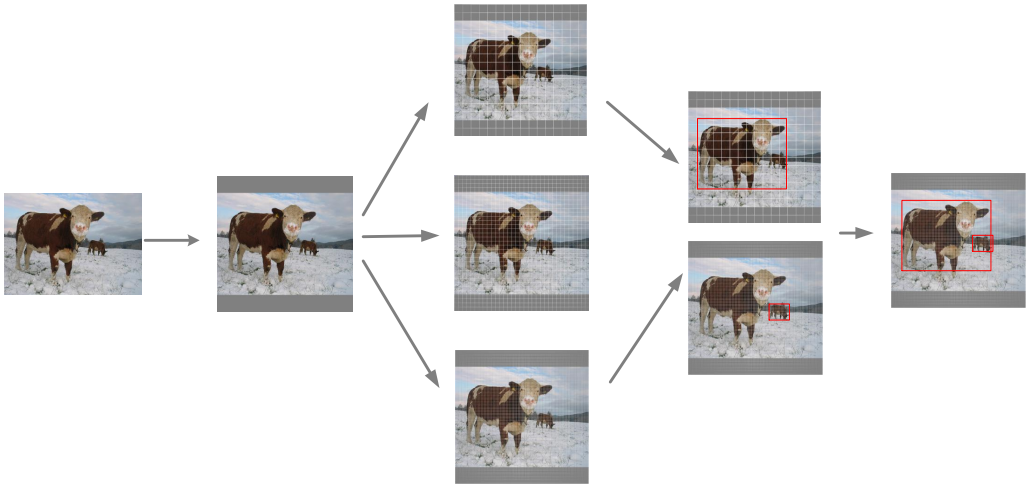

Figure 1. YOLOv4 detection grid division diagram.

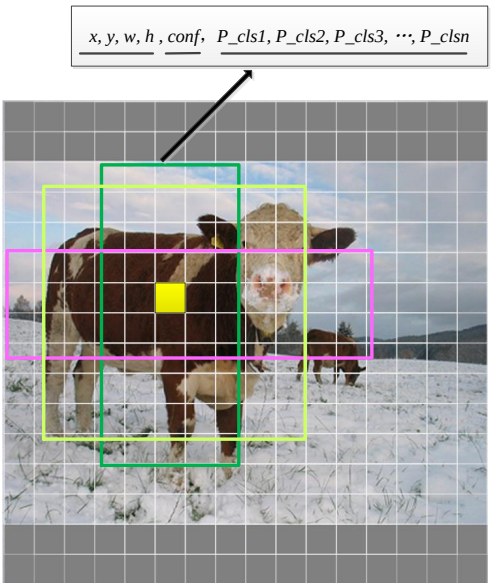

Figure 2. YOLOv4 prior box output example.

2.2. Physical-World Object-Detection Adversarial Attack

In 2017, Brown first proposed the concept of adversarial patches and focused the attack on a patch independent of the image, thus successfully misleading CNN classifiers [18]. Subsequently, researchers applied this idea to the object-detection adversarial-attack task and carried out further research following its principles and characteristics.

Pedestrian detection is widely used in computer applications, such as vehicle-assisted driving, motion analysis, etc. Ref. [20] is the first to propose an adversarial patch that generates an intra-class variation attack, enabling pedestrians to evade the YOLOv2 detector. Based on this, Ref. [21] proposed a TPS non-rigid transformation and checkerboard mapping method to successfully print the generated adversarial patch on the T-shirt. In this way, when people walk around in clothes, the twisted T-shirt can still evade the object detection model. Ref. [27] added a frequency attention module to improve the attack effect of small- and medium-sized patches, and the baseline method of the literature was adopted in the attack algorithm. Later, researchers began to focus on generating more natural adversarial patches after implementing pedestrian-detection attacks in the physical world, and Ref. [28] proposed a universal physical camouflage attack for the wild. Ref. [29] designed physical adversarial patches for object detectors by utilizing image manifolds learned on real-world

images by pre-trained generative adversarial networks, and natural-looking adversarial patches were generated by sampling optimal images from GANs. Ref. [30] also designed a legal adversarial patch that looks more realistic to the naked eye, using animated images as a starting point for patches, and this work proposed a new framework for a two-stage training strategy to combat patches. Ref. [22] put forward the concept of adversarial texture, and based on the adversarial T-shirt, a scalable generative attack $(TC\text{-}EGA)$ method with torus clipping was proposed to make AdvTexture have a repetitive structure. *TC-EGA* can evade human detectors from different viewing angles, and it still uses the baseline confidence attack algorithm as the basic attack algorithm.

Compared with the fixed-shape attack of autonomous driving stop signs and the frontal attack in face recognition, the pedestrian detection task is more difficult due to the variety of pedestrian postures and complex scenes. In recent years, with the deepening of research, the naturalness and legitimacy of adversarial patches in the physical world have been studied more deeply, bringing a series of innovations, such as multi-dimensional angle confrontation tops, and T-shirts with adversarial animated images. Although the current adversarial attacks are better in terms of the consideration and development of physical factors, they all adopt a single initialization method based on Ref. [20], and use random noise or grayscale images as the starting point for adversarial patch training, without considering initial diversification. It can be seen from Ref. [31] that the diversity of input space does not lead to the diversity of output space, and this paper believes that a single initialization method limits the understanding space during the adversarial patch generation process. Therefore, this paper proposes an initially diverse general adversarial patch attack method, which aims to further improve the object-detection-attack performance of the baseline method.

3. Improved Universal Adversarial-Patch-Diversity Initialization Attack Algorithm

In this section, a diverse and diversified adversarial patch generation method is proposed for the problem of the singleness of the initialization direction in the traditional adversarial-patch-generation process, which draws on the principle of output diversified sampling (ODS) [31] to provide a more effective and diverse starting point for attacks. This paper combines output diversification initialization (ODI), following the principle of object detection to generate stronger adversarial patches to attack the object detector.

The following describes the generation principle of the object detection adversarial patch: initialize the adversarial patch block, paste it at the target location, set the corresponding attack loss function, and optimize the adversarial patch to make it aggressive by the method based on gradient backpropagation, thereby generating an adversarial patch that can evade the object detection model. Different from the existing adversarial patch initialization method, this paper uses a diversified initial method instead of the existing random initialization adversarial patch block.

The process is shown in Figure 3, and the details are shown below:

1. Initialize the adversarial patch. First, an n $\times$ n $\times$ 3 diversified patch is generated by using the ODI algorithm, where n is the image size and 3 is the image channel.
2. Patch transformation. Performs a variety of random transformations, including rotation, cropping, adding noise, and shading changes on the generated adversarial patch to improve the robustness of adversarial patch training.
3. Paste the patch. Determine the target position in the image according to the dataset label, and then place the converted adversarial patch in the center position of the person in the graph for subsequent training.
4. Set the loss function. According to different attack tasks, set different loss functions, including target category loss, target positioning loss, and target confidence level.
5. Enter the detection model. Construct an object detection model, and input the generated image data with adversarial patches into the detection model for detection and localization classification.

6. Gradient backpropagation. Using the Adam optimizer, perform iterative training to update the data of the adversarial patch in the image through the backpropagation algorithm until the training reaches the epoch round or loss convergence, and the adversarial patch image is obtained.

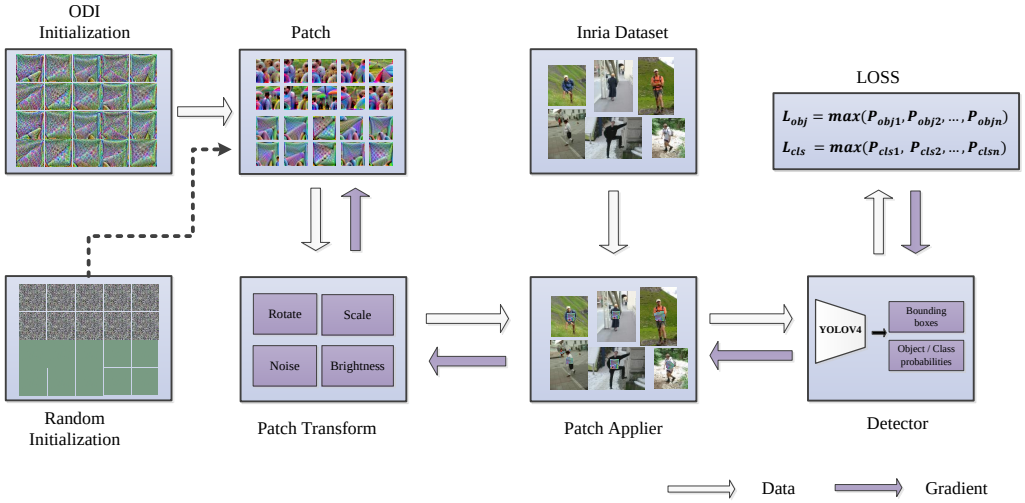

Figure 3. The overall design of the adversarial patch generation.

3.1. Analysis of the Principle of Attack Algorithm

The adversarial patch differs from the global perturbation attack category in the original paper, and the object detector has a more complex task and network model. However, in the process of generating adversarial patches, this paper argues that the adversarial patches have the same characteristics, i.e., the diversity of random initialization cannot be mapped to the diversity of the output space. To this end, this paper adopts the output diversity method of multiple restarts to combat patch generation. Specifically, the initialization diversity algorithm after the model output is designed as follows:

$$v_O DI = \frac{\bigtriangledown_x \left(w_d^T f(x) \right)}{\| \bigtriangledown_x \left(w_d^T f(x) \right) \|_2} \tag{1}$$

where $w_d[-1,1]$ is the initial diversification direction, sampling from the uniform distribution over $[-1,1]$, and $f(x)$ is the classifier model output. Due to the complex diversity of object detection tasks, the randomly generated w_d from the uniform distribution of $[-1,1]$ in the original paper cannot meet the requirements of the initial diversified functions of the object detection task. According to the specific attack task and optimization target, this paper reselects an appropriate w_d. The specific reasons why the uniform distribution of $[-1,1]$ cannot satisfy the task and how to obtain it are introduced in Section 3.2. The improved attack algorithm diversity–universal adversarial patch (D-UAP) used in this paper is shown in the following.

The Algorithm 1 uses a stochastic method to generate the initial adversarial patch and applies it to the dataset after random transformation. Then, the formal training process begins, which iterates by using the diversified initialization method, adds the corresponding generated diversified direction at the model output, and takes the updated adversarial patch as a new starting point; after the initialization is completed, the target model is attacked by the original attack method, and finally, the attack patch after training is obtained. Due to the randomness of the w_d direction selection, using a single random may not necessarily find an effective initialization direction. As shown in Figure 4, taking the spheroids as an example, different selection cases of the initial diversification direction and the solution space they fall into are observed, and the new direction can lead to the sub-optimal solution

of the model. Therefore, this paper combines the multiple restart mechanism to make multiple selections of random directions to find a better initialization direction.

Algorithm 1 D-UAP Algorithm

Input: the original image x_org, the object detector D, the patch transformation function $Transform$, the patch application function $Applier$, optimizer $Adam$, the D-UAP number of restarts $restart$, the diverse step size N_odi, the attack number of iterations s, the learning rate l_r, the diversity direction parameter w_d.

Output: adversarial patch $patch_adv$

Set $w_d \sim U(-a, b)$, $patch_0 = random()$
for r in $restart$ **do**
  optimizer = $Adam(patch_0, l_r)$
  for i in N_odi **do**
    $patch_i = Transform(patch_i)$
    $x_i = Applier(x_org, patch_i)$
    $v_{ODI} = v_{ODI}(x_i, D, w_d)$
    Updata $patch_{i+1} using optimizer, v_{ODI}$
  end for
  for j in s **do**
    $patch_j = Transform(patch_j)$
    $x_j = Applier(x_org, patch_j)$
    Update $patch_{j+1}$ using optimizer
    $patch_adv = clip(patch_{j+1})$
  end for
end for
return $patch_adv$

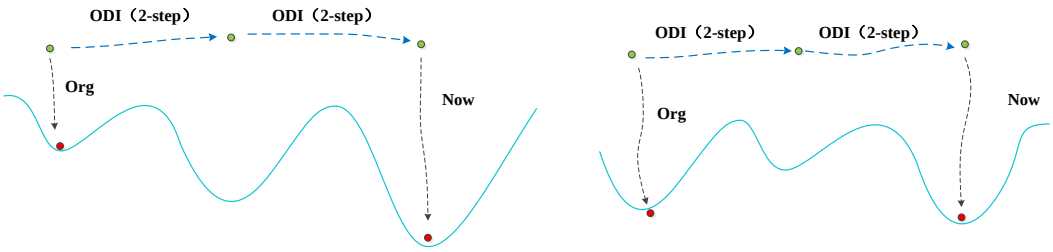

Figure 4. Initial diversification direction example.

To determine the optimal initial number of diversified steps, this paper carries out the ODI step size selection experiment. Let N_odi denote the diverse step size; according to reference [31], this paper sets $N_odi[2, 4, 6, 8, 16]$. Meanwhile, the total training round is set to 50, the number of *restart* is set to 100, the diversity of loss space changes is observed by statistical means, and finally, the N_odi with the largest change in the loss space is selected as the diversified initial step size used in this paper.

Taking the OBJ attack as an example, this paper conducts experiments to explore the influence of N_odi step selection on the diversity of the loss space, thus obtaining the optimal step selection.

Shown in Figure 5, *loss_odi i* is the loss space chart at the time when the step count is i, and it can be observed from the first row that with the increase in the initial number of diversified steps, the loss space diversity gradually increases, i.e., its distribution range is increasingly extensive. The second line *Loss_epoch* 50 in Figure 5 corresponds to the loss space change statistical chart after 50 rounds at the i moment. It can be observed that with the increase in the initial diversification of steps, it will not continue to bring about the diversification of the loss space. As shown in the Figure 5, when the initial number of steps N_odi is 4, the statistical loss value distribution range is the largest, and the loss space

diversification reaches the maximum. Therefore, this paper chooses the N_odi of four as the initial diversification step in this paper.

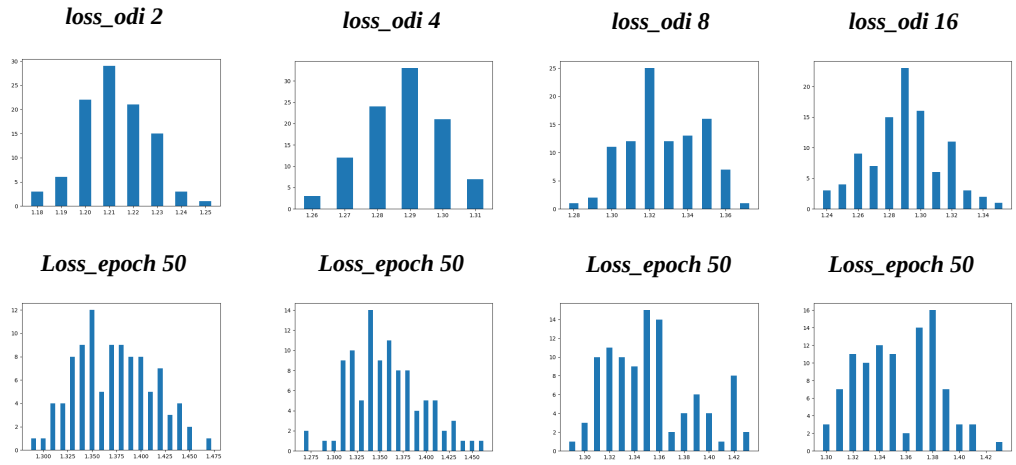

Figure 5. The loss function distribution of different N_odi parameters and the corresponding 50 epochs.

Then, a parameter sensitivity analysis experiment is conducted to verify the influence of different parameters on the attack effect, and the results are shown in Table 1.

Table 1. Sensitivity comparison chart.

N_odi	Mean	Maximum	Minimum
2	51.47%	56.45%	47.27%
4	51.75%	55.83%	45.87%
8	53.06%	56.14%	47.90%
16	51.59%	55.20%	48.05%

Through experiments, it is proved that D-UAP is sensitive to the N_odi parameter, the maximum difference between the values of different parameters can be 2%, and the difference between the maximum and minimum values of the unified parameters is about 10%. This is because different initialization directions have different superposition effects on attacks, and when choosing a good initialization direction, the attack effect of the optimized target can be greatly improved. According to the above result, when N_odi is set to 4, it has the best effect; under the comprehensive consideration of the calculation time, N_odi is set to 4, which not only contributes to better results, but also helps to find a diversified starting point.

According to the characteristics of the ontology and the expression of the quintuple set, this section designs a security ontology for defense strategy recommendations and lays an adaptable intelligence foundation for defense strategy selection. The security ontology for early attack warnings provides a formalized expression of its knowledge and can be exploited to perform reasoning and enforce security policies.

3.2. w_d Parameter Selection

Due to the particularity of the object detection task, the use of the above output diversified sampling method will simplify the attack, which is caused by the difference between the principle of the object detection model and its image classification model. The object detection model uses a preset prior box anchor for object detection. The prior box refers to the preset box of different sizes and different aspect ratios on the image, and the boxes of different sizes are set to obtain a larger intersection ratio so that there is a higher probability of better matching detection frames. In this paper, k-means clustering

is used to obtain the size of the prior box, in which there are a total of nine anchors of different sizes and aspect ratios. The output of each layer of the object detection model has four dimensions $[batch\_size, anchor\_num(5 + n_classes), h, w]$, and the a priori box corresponds to $(5 + n_classes)$ parameters. The first five bits are target position parameters, where bits 0–4correspond to the four adjustment parameters of the target location coordinates (tx, ty, tw, th), and the 5th bit is the confidence conf of the corresponding target of the detection box; bits 5–85 represent the probability that the target belongs to each category. Additionally, h and w correspond to the width and height of the mesh, and the corresponding sizes of the object detection model used in this paper are 13×13, 26×26, and 52×52, respectively.

Suppose that there is an image with a single-digit target number, and the number of detection boxes output by the model is 10,647 as shown in Figure 2. That is, $anchor\_num \times \sum_{i=0}^{3}(h_i \times w_i) = 3 \times (13 \times 13 + 26 \times 26 + 52 \times 52)$. However, according to the concept of positive and negative samples of YOLOv4, only the prior box greater than the IOU threshold is used as positive samples and the rest as negative samples. This shows the distribution of positive and negative samples and their unevenness in object detection. It is known that when an output is considered a negative sample, its output $conf$ confidence position is negative, i.e., under the normal model output, only a small number of $conf$ is positive. In this experiment, an image is tested to obtain negative samples of tens of thousands of digits and positive samples within two digits. In this case, if w_d still uses the value obtained from the uniform distribution of $[-1, 1]^C$, only a small number of output $conf$ will be negative, and a large number of output $conf$ will be positive. Combined with the object detection principle and the target location attack algorithm $det_{loss} = mean(max_{pro})$, max_{pro} is the maximum confidence of the category "person", the maximum confidence of max_{pro} is always 1, and in the same way, det_{loss} is always the same, i.e., the gradient update is always unchanged. Therefore, if w_d is not set correctly in this paper, the original w_d can be still used, and its function of initializing diversity is lost.

In this paper, the value of w_d is reselected according to the positive and negative sample ratio of the model, and the problem is corrected by redistributing the positive and negative values in the w_d. If the total number of samples output by the dataset is $n_samples$, and the model detects that the number of positive samples is $n_{positive}$, the proportion of positive samples is $rate = \frac{n_{positive}}{n_{samples}}$. The value of w_d is selected from the uniform distribution of $[-rate, +1]$ as shown below.

According to the above Algorithm 2, the $Rate_output$ is calculated to be 0.004. Thus, when the attack category is an object, the corresponding position of w_d in this document is obtained from the uniform distribution of $[-0.004, +1]$. When the attack category is cls, it is a category attack, which has the same principle as image classification. In this case, the corresponding position is obtained from the uniform distribution of $[-1, +1]$. When the attack category is object∗class, w_d is obtained from the uniform distribution of $[-0.004, +1]$ at the $conf$ position, and from the uniform distribution of $[-1, +1]$ at the class position.

Algorithm 2 Adversarial attack method based on output diversification initialization

Input: model *Model*, dataset *Dataset*, number of prior bounding boxes per grid *anchor_num*, target confidence $conf$
Output: positive samples rate $Rate_output$

```
Set rate_v = []
for image in Dataset do
    output = Model(image)
    for index in range (len(output)) do
        for j in s_2 do
            Get the target confidence conf from the corresponding location in the model output
            n_positive = ∑(conf > 0)        ▷ Get the total number of elements in the conf array > 0
            Get n_all of the elements in the conf array
            rate = n_positive / n_all
            rate_v.append(rate)
        end for
        Rate_output = ∑ rate_v / len(rate_v)
    end for
end for
return Rateoutput
```

3.3. The Loss Function Part of the D-UAP Method

To improve the physical utility of the adversarial patch, in the study on adversarial glasses [32], the authors proposed NPS (*non-printability-score*) and total variation. Our work follows in the footsteps of Thys [20] in 2019 and continues to use these two utility loss functions.

Given that the computer RGB color space is P, the colors that printers and other devices can copy and print are only a subset of the computer color collection. Therefore, to print more robust adversarial patches, we need to make as many adversarial patches as the printers can print. Previous studies defined the non-printability score NPS and set 30 RGB color combinations that can be printed as the optimization direction [32]. If the set of colors that can be printed is $C \subset P$, then the NPS score for the unprintability of a pixel $\hat{p}$ is

$$NPS(\hat{P}) = \min_{cprint \in C} \left| \hat{P} - cprint \right| \tag{2}$$

where $cprint$ denotes a set of 30 printable colors in C, and $\hat{p}$ denotes our RGB colors against the patch. $\hat{p} \in P$, and when $\hat{p}$ is closer to $cprint \in C$, the NPS score is smaller; in this case, it is the most likely to print an adversarial patch. Therefore, non-printability is taken as one of our optimization goals, and the non-printability loss function is as follows:

$$L_nps = \sum_{\hat{p} \in Patch} NPS(\hat{p}) \tag{3}$$

where Patch is an adversarial patch, and $\hat{p}$ is a pixel in the adversarial patch.

To improve printability, studies have shown that natural images are smooth and consistent during capture, i.e., changes between pixels are smooth, and non-smooth adversarial patches may not be attacked in practice. Therefore, to find the perturbation of smooth consistency, the total amount of change in the image is set to L_tv, and the smoothing loss function is set to ensure a smooth color transition [33]. L_tv is expressed as follows:

$$L_tv = \sum_{i,j} \sqrt{\left(\left(r_{i,j} - r_{i+1,j} \right)^2 + \left(r_{i,j} - r_{i,j+1} \right) \right)^2} \tag{4}$$

where (i, j) is the pixel at the coordinates (i, j) in the adversarial patch. When the values of neighboring pixels are similar, L_tv is smaller, and the adversarial image is smooth. Therefore, to better implement attacks in the physical world, this paper also takes the smoothness loss function as one of the optimization goals.

The attack target is shown in Figure 3. Let the object detection loss function be L_det. When the target is the object detection bounding box, $L_det = L_obj$; when the target of the attack is the target category, $L_det = L_cls$; when the attack task is a combination of both, $L_det = L_obj * L_cls$. The optimization goal in this paper is represented as follows:

$$L = \alpha * L_nps + \beta * L_tv + \gamma * L_det \tag{5}$$

where α, β and γ are the weight coefficients.

4. Design of Experiments

This section provides the detailed configuration of the experiment in this paper. Firstly, the experimental environment is introduced, mainly including software and hardware settings, as well as the dataset used and the configuration parameters of the attack model. Secondly, the details of the experiment are given, and the parameter selection in the specific experimental process is described.

4.1. Experiment Configuration

The experiments are conducted on a server equipped with an Intel(R) Xeon(R) Gold 5218R CPU @ 2.10 GHz, 125.5 GB main memory, and an NVIDIA GeForce RTX 3090Ti GPU with 24 GB video memory. The algorithm is implemented with the PyTorch deep learning framework.

The dataset used in this study is the INRIA pedestrian dataset, which contains 614 positive samples in the training set and 1273 pedestrians; the test set contains 288 sheets and 589 pedestrians. The dataset shows that most of the people are in a standing position and are taller than 100 pixels. This helps to place adversarial patches during training, so the dataset is more suitable for this study than PASCAL VOC and MS COCO.

The YOLOv4 object detection model is taken as the attack object model. Referring to the literature [19], the non-maximum suppression threshold is set to 0.4, and the IOU threshold is set to 0.5. To obtain more samples for training, this paper sets the confidence threshold to 0.25, and better samples are selected to participate in the training process.

4.2. Implementation Details

In this experiment, the diversified initial step size introduced in Section 3.1 is investigated, and according to the analysis of the loss function in Section 3.3, three sets of experiments are set up, corresponding to category attacks, confidence attacks, and category plus confidence attacks. In this paper, the number of restarts for each group is set to 10 times. Since w_d is randomly generated during each restart, 10 different adversarial patches are obtained. In this experiment, the adversarial patch generation method [20] in the original paper is taken for comparison, and the training epoch of both groups of experiments is set to 600 rounds.

To verify that the initial diversification attack can generate countermeasures faster and the generated countermeasures have stronger attack effects, the ODI step size in Section 3.1 is selected as the parameter of this experiment. The number of restarts of each group of training is set to 10, 10 different initialization directions w_d are randomly generated, and the experimental result is compared with that of the original adversarial patch generation method [20]. According to the analysis of the loss function in Section 3.3, the comparative experiment is divided into three groups—category attacks, confidence attacks, and category plus confidence attacks—in which the training epoch is uniformly set to 600 rounds.

5. Experimental Results

In this paper, recall is adopted as the evaluation index. In the following table, YUAN represents the original adversarial patch attack method [20], and ODI ATTACK represents the diversified initial direction of the adversarial patch attack method proposed in this paper. The detection recall of the original model is taken as the baseline, and according to the attack target, three sets of comparative experiments are carried out.

As shown in Table 2, the adversarial patch generation method proposed in this paper achieves better attack effects than the original method for the three attack targets, among which the OBJ attack is the best, and the recall is reduced by 8.46%, compared with the original adversarial patch attack.

Table 2. Comparison of the effects of the recall attack.

Approach	**Recall**	
Clean	100%	
	YUAN	ODI_{ATTACK}
$OBJ * CLS$	53.18%	48.83%
OBJ	52.41%	45.87%
CLS	93.15%	86.93%

Figure 6 shows the attack effect of the proposed method, where the horizontal images correspond to the clean image, the detection effect of the clean image, and the detection effect of the adversarial patch image generated by the superimposed method in this paper, and the vertical images correspond to the image of one person to a large number of people. According to the detection effect, when the adversarial patch proposed in this paper is performed on the image, the YOLOv4 detector can detect the person in the image well. However, when the adversarial patch is superimposed on the person, the detection performance of the detector is greatly reduced, and only a few people can be detected by the detector, which proves the effectiveness of the attack method proposed in this paper.

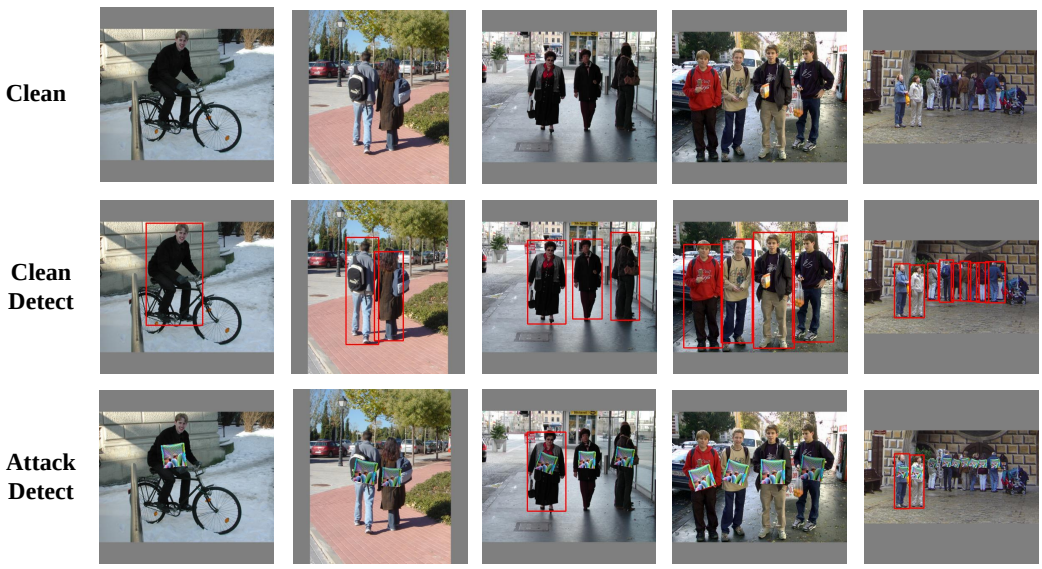

Figure 6. Comparison of detection results.

5.1. Comparative Analysis of Training Epoch

This section analyzes the training time overhead of the proposed adversarial patch generation method and the original adversarial patch generation method, in which the *TV* loss is the smoothness loss function, NPS loss is the non-printability loss function, the DET

loss refers to the OBJ loss function in Section 3.3, and recall is the recall rate. These values adopt normalized reality.

Figure 7 shows the loss graph of the classic adversarial patch generation method [20] during the training process. It can be seen that the training loss converges at about 700 rounds, indicating that the original attack method needs more rounds to achieve convergence to complete the training process.

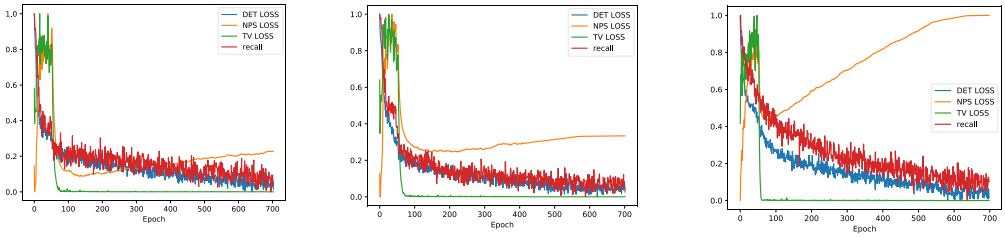

Figure 7. The loss graph of the classic training method.

Figure 8 shows the loss training graph of the OBJ attack in this paper. It can be seen that in the 10 groups from 0 to 9, except for the second group, which converges slowly, the rest of the groups converge in 300 to 400 training rounds. Using the attack method proposed in this paper, adversarial patches can converge in a short time, which greatly reduces the complexity of the algorithm and the time overhead of generating adversarial patches.

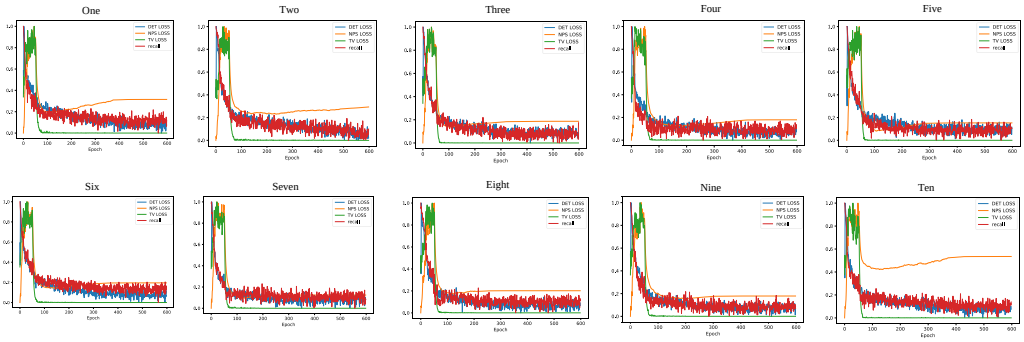

Figure 8. The training loss graph of the D-UAP method.

5.2. Physical World Attack Display

To verify the aggressiveness of the adversarial patch generated in this paper in a real scenario, the generated adversarial patch is printed for the YOLOv4 detection model; two sets of targets are selected for testing, and the test results are illustrated in Figure 9.

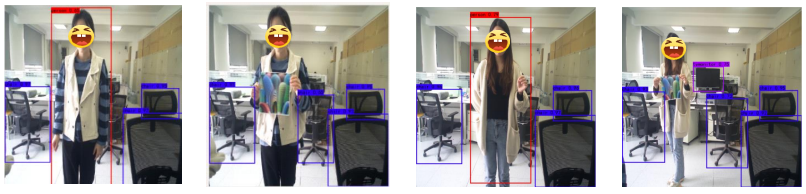

Figure 9. The detection graph of the attack in the physical world.

The above figure corresponds to normal object detection and object detection with the adversarial patch, respectively. It can be seen that the chair can be correctly identified in both images, and only the character carrying the adversarial patch successfully evades the detection of the detector. Thus, the two sets of examples verify the effectiveness of the proposed adversarial patch attack method in the real world.

Figure 10 shows the comparison of the person carrying and not carrying the adversarial patch. It can be seen that when there is no adversarial patch, the target person standing

normally and carrying a book can be correctly identified; after carrying the adversarial patch, the detector fails to detect the presence of the target person.

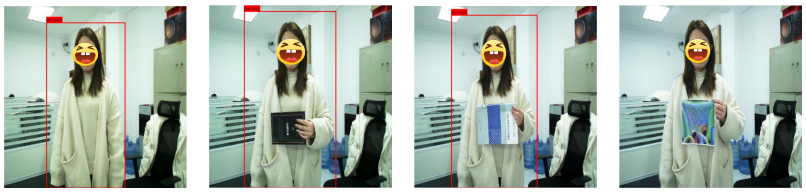

Figure 10. Comparison graph of person detection in holding a book.

5.3. Complementary Experiments

The above analysis proves that the adversarial patch can evade the detector, but to further illustrate the effectiveness of the adversarial example, this paper selects multiple sets of patch pictures and pastes them to the target person in the same way for detection. This demonstrates that the evasion effect of generating the adversarial patch on the detector is not caused by the occlusion of the patch block but by the perturbation generated by the specific generation on the adversarial patch. As shown in Figure 11, random noise, a cartoon image, a flower image, and two adversarial patches generated by the YOLOv2 model are taken as test patches.

Figure 11. Test the patches.

Taking the above images as a control experiment, this paper applies these patches to the same image at the same time, takes multiple images as examples, adds different patches to the example images, inputs them into the model for detection, and then investigates the detection effect of the detector on these images. The following figure shows the detection effect of YOLOv4 on images with different patches.

It can be seen from Figure 12 that the use of random noise countermeasure patches and cartoon images cannot achieve the aggressiveness of the adversarial example generated in this section to the detection model, which proves the particularity of the adversarial example. Although individual images such as the third row and third column of images in the obscuration of the flower patch can make the model fail to correctly identify the target, this situation is rarer than the object detection adversarial example. Unlike other specific pictures, the adversarial example can decrease the performance of the detection model.

In this section, the random adversarial patch comparison experiment is added, and recall is still selected as an evaluation index, taking the OBJ attack as an example, and the test results are as follows:

As can be seen in Table 3, the model's detection performance of the dataset is relatively degraded after adding noise and other network pictures, but the overall detection rate is good, and most of the targets can be identified. In addition, the recall values of the two adversarial patches trained on the YOLOv2 model on YOLOv4 show that the object detection adversarial examples have a certain degree of mobility to the detection model, and compared with other random patterns, the carefully trained adversarial examples still have a greater impact on the detection model than the random patches.

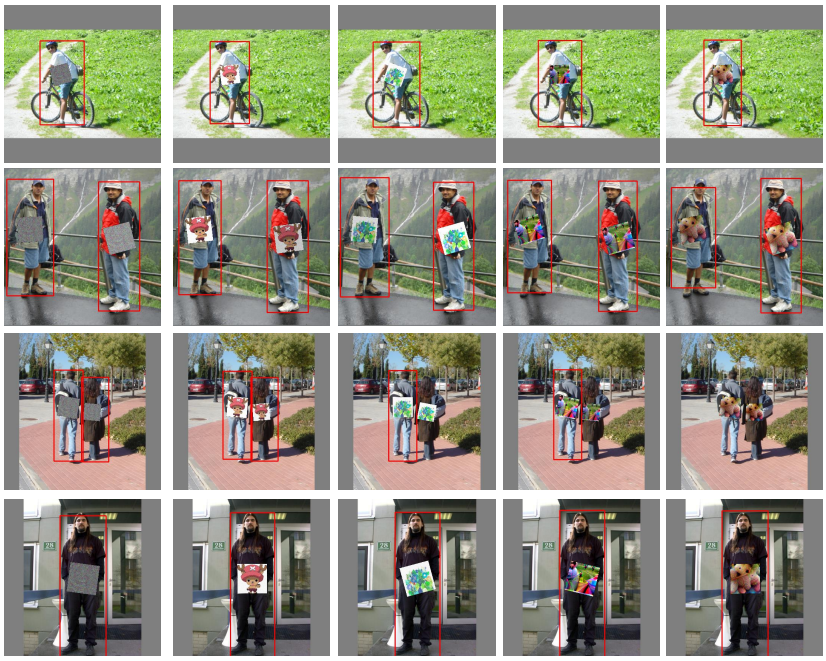

Figure 12. Detection effect with patch.

Table 3. Compare to the patch to detect recall.

Patch	Recall
Clean	100%
Noise	89.89%
Cartoon	88.18%
Flower	88.33%
YOLOv2-1	74.02%
YOLOv2-2	82.27%
OURS	45.87%

6. Conclusions

This paper proposes an initial diversified generation method for generating adversarial patches, which reconsiders the impact of the adversarial initialization direction on its training results based on the traditional adversarial patch generation mechanism. Aiming at pedestrian detection attacks, the statistics-based initial diversified step count N_odi is combined with restart training, and the w_d random direction is combined with simultaneous training of the adversarial patch with multiple different starting points.

In this way, the adversarial patch jumps out of the limited solution space and obtains a better solution. Finally, the effectiveness of the proposed method is proved by experimentally showing that the adversarial patch trained from a new starting point has a larger *logist* space and stronger aggression than that trained from the same starting point. By using the diverse adversarial patch training methods proposed in this paper, combined with existing physical clothing simulation technology, it is possible to obtain better adversarial T-shirts than the existing methods, so pedestrians can better avoid pedestrian detectors.

Author Contributions: Conceptualization, X.W. and L.S.; methodology, Y.Y. and X.W.; validation, X.W. and Y.Y.; formal analysis, X.W. and L.S.; investigation, L.S. and X.W.; resources, Y.Y. and X.W.; writing—original draft preparation, X.W. and Y.Y.; writing—review and editing, X.M. and L.S.; visualization, X.W. and Y.Y.; supervision, L.S. and X.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: Data openly available in a public repository.

Acknowledgments: We acknowledge the equal contribution of all the authors.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Chui, K.T.; Gupta, B.B.; Chi, H.R.; Arya, V.; Alhalabi, W.; Ruiz, M.T.; Shen, C.W. Transfer learning-based multi-scale denoising convolutional neural network for prostate cancer detection. *Cancers* **2022**, *14*, 3687. [CrossRef] [PubMed]
2. Li, T.; Chang, H.; Mishra, S.; Zhang, H.; Katabi, D.; Krishnan, D. Mage: Masked generative encoder to unify representation learning and image synthesis. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 2142–2152.
3. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.
4. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
5. Wang, C.Y.; Bochkovskiy, A.; Liao, H.Y.M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 7464–7475.
6. Deng, J.; Guo, J.; Xue, N.; Zafeiriou, S. Arcface: Additive angular margin loss for deep face recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4690–4699.
7. Liu, W.; Wen, Y.; Yu, Z.; Li, M.; Raj, B.; Song, L. Sphereface: Deep hypersphere embedding for face recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 212–220.
8. Yang, X.; Liu, C.; Xu, L.; Wang, Y.; Dong, Y.; Chen, N.; Su, H.; Zhu, J. Towards Effective Adversarial Textured 3D Meshes on Physical Face Recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 4119–4128.
9. Gupta, D.K.; Arya, D.; Gavves, E. Rotation equivariant siamese networks for tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 12362–12371.
10. Yan, B.; Zhang, X.; Wang, D.; Lu, H.; Yang, X. Alpha-refine: Boosting tracking performance by precise bounding box estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 5289–5298.
11. Cao, J.; Pang, J.; Weng, X.; Khirodkar, R.; Kitani, K. Observation-centric sort: Rethinking sort for robust multi-object tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 9686–9696.
12. Viola, P.; Jones, M. Rapid object detection using a boosted cascade of simple features. In Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2001), Kauai, HI, USA, 8–14 December 2001; Volume 1.
13. Felzenszwalb, P.; McAllester, D.; Ramanan, D. A discriminatively trained, multiscale, deformable part model. In Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA, 23–28 June 2008; pp. 1–8.
14. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
15. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28*. [CrossRef] [PubMed]
16. Xie, C.; Wang, J.; Zhang, Z.; Zhou, Y.; Xie, L.; Yuille, A. Adversarial examples for semantic segmentation and object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 1369–1378.
17. Goodfellow, I.J.; Shlens, J.; Szegedy, C. Explaining and harnessing adversarial examples. *arXiv* **2014**, arXiv:1412.6572.
18. Brown, T.B.; Mané, D.; Roy, A.; Abadi, M.; Gilmer, J. Adversarial patch. *arXiv* **2017**, arXiv:1712.09665.
19. Li, Y.; Bian, X.; Chang, M.C.; Lyu, S. Exploring the vulnerability of single shot module in object detectors via imperceptible background patches. *arXiv* **2018**, arXiv:1809.05966.
20. Thys, S.; Van Ranst, W.; Goedemé, T. Fooling automated surveillance cameras: Adversarial patches to attack person detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Long Beach, CA, USA, 15–20 June 2019.
21. Xu, K.; Zhang, G.; Liu, S.; Fan, Q.; Sun, M.; Chen, H.; Chen, P.Y.; Wang, Y.; Lin, X. Adversarial t-shirt! evading person detectors in a physical world. In Proceedings of the Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Proceedings, Part V 16; Springer: Berlin, Germany, 2020; pp. 665–681.
22. Hu, Z.; Huang, S.; Zhu, X.; Sun, F.; Zhang, B.; Hu, X. Adversarial texture for fooling person detectors in the physical world. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 13307–13316.

23. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
24. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
25. Li, C.; Li, L.; Jiang, H.; Weng, K.; Geng, Y.; Li, L.; Ke, Z.; Li, Q.; Cheng, M.; Nie, W.; et al. YOLOv6: A single-stage object detection framework for industrial applications. *arXiv* **2022**, arXiv:2209.02976.
26. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; Sun, J. Yolox: Exceeding yolo series in 2021. *arXiv* **2021**, arXiv:2107.08430.
27. Lei, X.; Cai, X.; Lu, C.; Jiang, Z.; Gong, Z.; Lu, L. Using frequency attention to make adversarial patch powerful against person detector. *IEEE Access* **2022**, *11*, 27217–27225. [CrossRef]
28. Huang, L.; Gao, C.; Zhou, Y.; Xie, C.; Yuille, A.L.; Zou, C.; Liu, N. Universal physical camouflage attacks on object detectors. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 2020; pp. 720–729.
29. Hu, Y.C.T.; Kung, B.H.; Tan, D.S.; Chen, J.C.; Hua, K.L.; Cheng, W.H. Naturalistic physical adversarial patch for object detectors. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 7848–7857.
30. Tan, J.; Ji, N.; Xie, H.; Xiang, X. Legitimate adversarial patches: Evading human eyes and detection models in the physical world. In Proceedings of the 29th ACM International Conference on Multimedia, Virtual Event, 24–25 October 2021; pp. 5307–5315.
31. Tashiro, Y.; Song, Y.; Ermon, S. Diversity can be Transferred: Output Diversification for White- and Black-box Attacks. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 4536–4548.
32. Sharif, M.; Bhagavatula, S.; Bauer, L.; Reiter, M.K. Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition. In Proceedings of the 2016 ACM Sigsac Conference on Computer and Communications Security, Vienna, Austria, 24–28 October 2016; pp. 1528–1540.
33. Mahendran, A.; Vedaldi, A. Understanding Deep Image Representations by Inverting Them. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015.